Mineração de Dados em
Biologia Molecular
WEKA
André C. P. L. F. de Carvalho
Monitor: Valéria Carvalho

Tópicos
Introdução
Simuladores de MD
Ambiente WEKA
Principais módulos
André Ponce de Leon de Carvalho
2

Usando MD
Implementação e simulação
Escrever código do zero
Reaproveitar código
Utilizar um ambiente dedicado
Fornece ferramentas apropriadas para auxiliar o usuário ao longo de toda a simulação
Minimiza esforço do usuário
Agiliza implementação e realização de experimentos
André Ponce de Leon de Carvalho
3

Ambientes para MD
Principais tipos
Baseados em telas e interfaces gráficas
Baseados em biblioteca de funções (lego)
Baseados em uma linguagem de programação específica
Funcionalidades
Suporte a pré-processamento e análise de resultados
Grande número de algoritmos de AM, técnicas estatísticas e ferramentas de visualização
André Ponce de Leon de Carvalho
4

Ambientes de MD
Comerciais
SAS
SPSS
Matlab
Públicos
Rapid miner
Knime
WEKA
André Ponce de Leon de Carvalho
5

WEKA o Pássaro
Pássaro típico da Nova Zelândia
Encontrado apenas lá
Não voa
Tamanho de uma galinha
Alimentam-se de invertebrados e frutas
Dócil em extremo
Quase foram a extinção
André Ponce de Leon de Carvalho
6

# WEKA o Pássaro

*Copyright: Martin Kramer (mkramer@wxs.nl)*

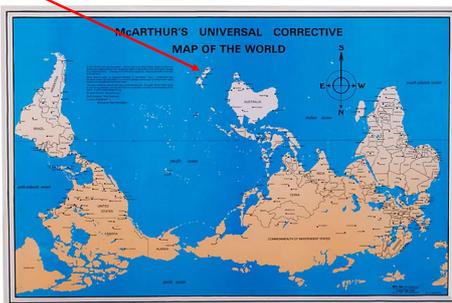


# Weka o Ambiente

- Desenvolvido na Universidade de Waikato, NZ
- Mineração de Dados e Aprendizado de Máquina
- Escrito em JAVA
  - Distribuído por meio da licença de software livre da GNU
- Utilizada com diferentes propósitos
  - Ensino, pesquisa, aplicações



# Onde fica Waikato?

McARTHUR'S UNIVERSAL CORRECTIVE MAP OF THE WORLD



# História do WEKA

- 1992:
  - Submissão do projeto para financiamento pelo governo da NZ por Ian Witten
- 1993:
  - Financiado aprovado pelo governo
  - Sigla WEKA criada por Geoff Holmes
    - Waikato Environment for Knowledge Analysis
  - Criação do formato ARFF por Andrew Donkin
    - **A**ttribute-**R**elation **F**ile **F**ormat
    - Segundo rumores, **A**ndrew's **R**idiculous **F**ile **F**ormat



# História do WEKA

- 1994:
  - Primeira versão é disponibilizada
  - Interface TCL/TK e algoritmos de aprendizado escritos principalmente em C
  - Versão beta
- 1996:
  - Primeira versão pública: WEKA 2.1



# História do WEKA

- 1997:
  - Decisão de reescrever o código em Java
    - Tese de doutorado de Eibe Frank
    - 1998
  - Disponibilizado WEKA 2.3, última versão baseada em TCL/TK
- 1998:
  - Disponibilizado WEKA 3
    - Primeira versão completamente em Java
    - Inclusão do módulo de desenvolvimento, com GUI
    - Compatível com livro do Wittel e Frank

## Últimas Versões do WEKA

- Várias versões foram desenvolvidas depois:
  - WEKA 3.2: "versão GUI", adiciona GUI (versão do livro usava linha de comando)
  - WEKA 3.3: "versão de desenvolvimento" com várias melhorias
  - WEKA 3.4: "versão do livro" compatível com segunda edição do livro *Data Mining*
  - WEKA 3.6: última versão estável (3.6.7)
  - WEKA 3.7: finalizada, mas ainda instável



## Ambiente WEKA

- Possui vários módulos, que podem ser utilizados para:
  - Pré-processamento de dados
  - Uso de algoritmos de Aprendizado de Máquina
  - Análise de resultados
  - Visualização gráfica de dados e modelos
  - Comparação de modelos (e algoritmos)



## Ambiente WEKA

- Com WEKA, é possível
  - Abrir um conjunto de dados em diferentes formatos
  - Pré-processar os dados por meio de filtros
  - Aplicar um algoritmo de AM aos dados
    - Ajustar parâmetros dos algoritmos
  - Analisar os resultados obtidos
    - Utilizar diversas medidas de desempenho



## Ambiente WEKA

- 3 formas de usar
  - Aplicar um algoritmo de aprendizado a um conjunto de dados e analisar a saída para saber mais sobre os dados
  - Usar modelos induzidos para gerar previsões para novos dados
  - Comparar desempenho de diferentes algoritmos para selecionar um deles



## Ambiente WEKA

- Como usar WEKA?
  - Linhas de comando
  - A partir de programas em Java
    - Chamar métodos do WEKA utilizando suas bibliotecas
      - Permite escrever e testar novos algoritmos
  - Usando GUI do Explorer
    - Mais fácil
    - Uma das opções do WEKA GUI *Chooser*



## GUI do WEKA

- *Graphical User Interface*
- Ponto inicial para chamar as principais aplicações e ferramentas de suporte
- Possui quatro opções:
  - *Explorer*
  - *Experimenter*
  - *KnowledgeFlow*
  - *SimpleCLI*

## GUI do WEKA

- Funcionalidades são acessadas por
  - Escolha de uma opção de um menu
    - Permite apenas as opções exibidas no momento da simulação
  - Preenchimento de valores em telas
    - Pré-preenchidos com valores *default* para obtenção de resultados com o mínimo de esforço
  - Entender os algoritmos e seu uso é importante para entender os resultados



## Principais Opções

- *Explorer*
  - Ambiente para exploração de dados
- *Experimenter*
  - Ambiente para realização de experimentos e testes estatísticos
  - Permite comparar modelos



## Principais Aplicações

- *KnwoledgeFlow*
  - Suporta mesmas funções que Explorer, além de aprendizado incremental
    - Permite trabalhar com fluxos de dados
  - Inclui interface *drag-and-drop* (arrastar e por)
- *SimpleCLI*
  - Interface para linha de comando
  - Permite que comandos do WEKA sejam executados por sistemas operacionais que não possuem interface GUI



## Explorer

- Ferramenta gráfica de fácil uso que mostra as funcionalidades do software
- Cada um dos principais pacotes disponíveis é representado na interface
  - Filtros
  - Algoritmos de classificação
  - Algoritmos de agrupamento
  - Algoritmos de associação
  - Algoritmos de seleção de atributos
  - Ferramenta de visualização



## Explorer

- Ferramentas de visualização permitem a análise em duas dimensões de:
  - Dados de um conjunto
  - Predições geradas por:
    - Modelos de classificação
    - Modelos de agrupamento de dados
  - Possui vários painéis

## KnowlegdeFlow

- Permite colocar ícones representando algoritmos e fontes de dados na tela
  - E uni-los na configuração desejada
- Permite definir um fluxo de dados
  - Conectando componentes representando
    - Fontes de dados
    - Ferramentas de pré-processamento
    - Algoritmos de aprendizado
    - Métodos de avaliação
    - Módulos de visualização



## Experimenter

- Ajuda a definir quando aplicar técnicas de classificação (e de regressão)
  - Que métodos e valores de parâmetros funcionam melhor para um dado problema?
    - Resposta não fácil
  - Permite comparar vários algoritmos de aprendizado
    - Pode ser feito interativamente com explorer
    - Módulo *Experimenter* automatiza esse processo



## Experimenter

- Como automatiza?
  - Torna mais fácil:
    - Executar algoritmos com:
      - Diferentes valores para seus parâmetros e
      - Vários conjuntos de dados
    - Avaliar desempenho estatístico
    - Realizar testes de significância
- Pode ser usado para distribuir carga em várias máquinas
  - Usando *Java Remote Method Invocation* (RMI)



## Documentação

- Documentação *online* gerada automaticamente do código fonte
- Única fonte completa dos algoritmos disponíveis
  - WEKA esta continuamente em crescimento
  - Por ser gerada automaticamente a partir do código fonte, documentação *online* esta sempre atualizada



## Prática

- Entrar na ferramenta WEKA
  - Selecionar opção explorer
  - Carregar base de dados iris.arrf
    - Opção "*Open file*"
  - Carregar classificador J48
    - Opção "*Classifier*"
    - Em *Classifier*, opção "*Choose*" J48
  - Realizar experimento usando training set
  - Olhar resultados



## Perguntas